Parveen Khurana

124 Followers About Following

Basics: Probability Theory

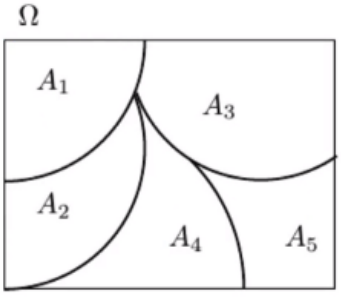
P Parveen Khurana Jan 4, 2020 · 9 min read

This article covers the content discussed in the Probability Theory module of the Deep Learning course and all the images are taken from the same module.

Introduction

The probability of any event A is always ≥ 0 and it will always be ≤ 1. So, probability values lie between 0 and 1 and that's the intuition behind using the output of Sigmoid Neuron as the probability value.

And if we have '**n**' disjoint events, the sum of the probability of the union of those events is equal to the sum of the probability of individual events.



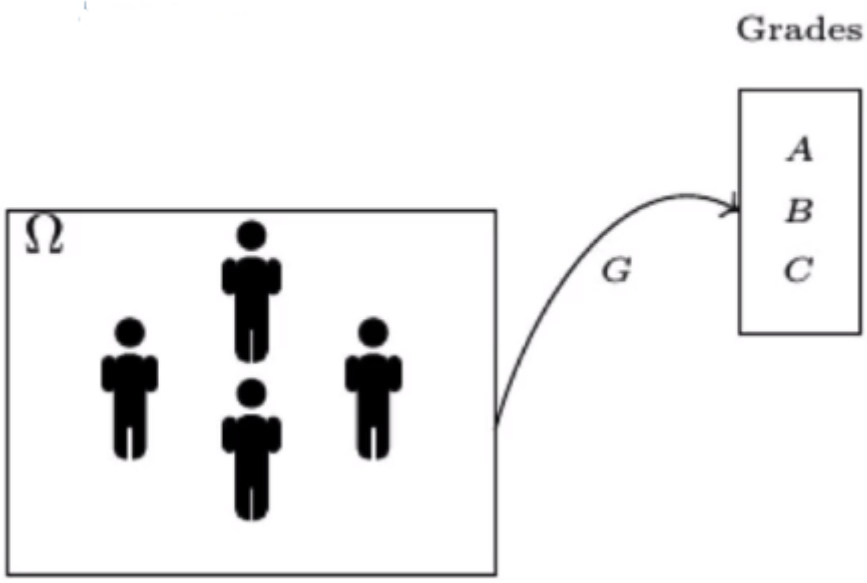
- For any event A,
$$P(A) \geq 0$$
- If $A_1, A_2, A_3, ..., A_n$ are disjoint events (i.e., $A_i \cap A_j = \phi \quad \forall i \neq j$) then
$$P(\cup A_i) = \sum_i P(A_i)$$
- If Ω is the universal set containing all events then
$$P(\Omega) = 1$$

Random Variable

student gets an ‘A’ grade or ‘B’ grade or ‘C’ grade. And now we can ask questions like what is the probability of a student getting an ‘A’ grade. The way we compute this is just the number of students with an ‘A’ grade divided by the total number of students.

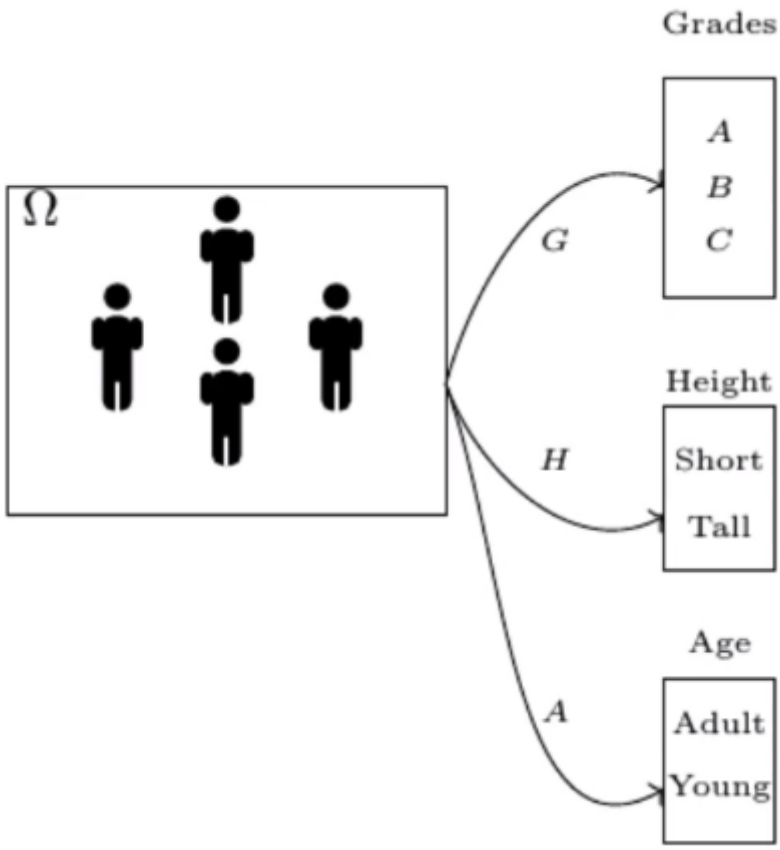
We could have another context as well, for example, we can have two events which tell whether the student is Tall or Short. And we compute the probability of this in the same way as discussed above for Grade. So, that’s one way of looking at this situation.

The other way of looking at this situation is something known as **Random Variable. Here, instead of looking at these events as separate, we could think of it this way as all the students form a set and now there is a mapping for each of the students in the set to one of the three possible grades.**



And we can think of this as a function. The function takes the student as the input and maps it to some grade and return the grade. And this function is actually called a random variable. The way we should look at it is that we have a set, every element of the set is mapped to some outcome or some value using this random variable.

Now the good thing about this view is that we can think of the students as a set and there is one function that maps students to Grades, another function maps students to



Now again we could ask questions like we have this random variable G and we are interested in the outcome of this random variable as equals to ‘A’. So, we are interested in the Probability of Grade equals ‘A’.

P(Grade = ‘A’)

- Suppose a student gets one of 3 possible grades in a course: A, B, C
- One way of interpreting this is that there are 3 possible events here
- Another way of looking at this is there is a random variable G which maps each student to one of the 3 possible values

where $g \in \{A, B, C\}$

Of course, both the interpretations of looking at the scenario are conceptually equivalent.

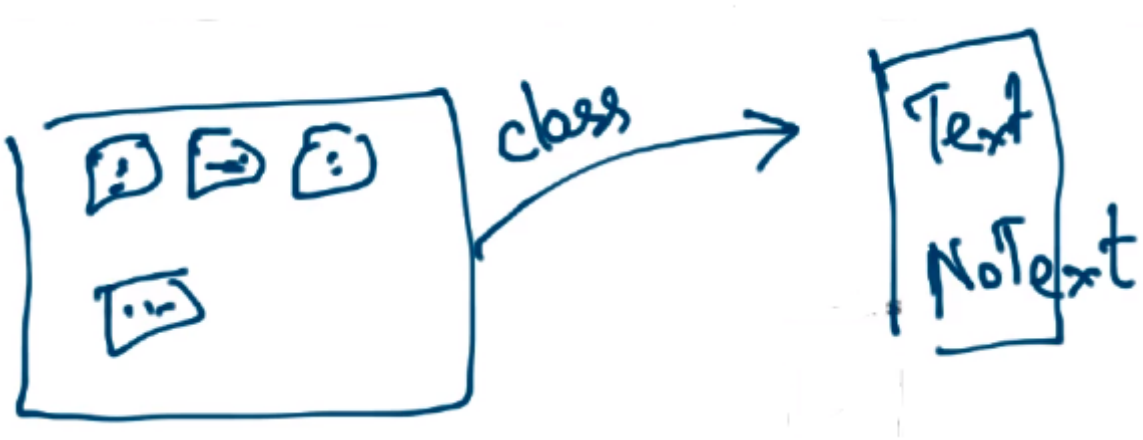
- But the second one (using random variables) is more compact
- Specifically, when there are multiple attributes associated with a student (outcome) - grade, height, age etc.
- We could have one random variable corresponding to each attribute
- And then ask for outcomes (or students) where $Grade = g$, $Height = h$, $Age = a$

The event **Grade = A**, we can look at this event as for all the elements belonging to the universal set which contains all the elements, such that when we apply the function Grade to them, then the answer is 'A'.

- A random variable is a function which maps each outcome in Ω to a value
- In the previous example, G (or f_{grade}) maps each student in Ω to a value: A, B or C

$$\text{event}\,\{\omega \in \Omega : f_{Grade} = A\}$$

The reason why we care about Random Variable is that in our capstone project, we can consider all the images as belonging to a set and now we can have a random variable that takes one of these images and tell if that image contains text(we can assign it a value of 1 if it contains text else a value of 0) or not.



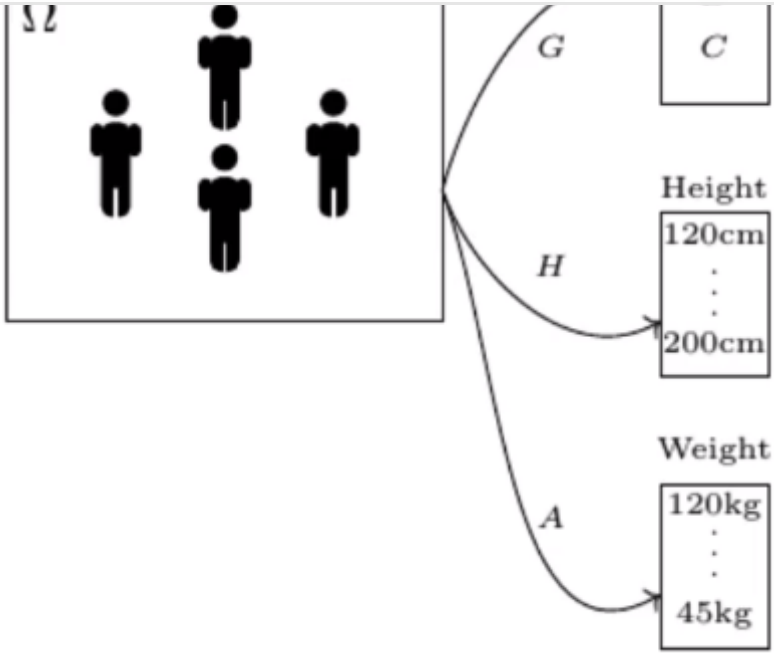
And as discussed in the above cases, we can have a random variable for Binary classification as well as for Multiclass classification. **Once the random variable gives us the probability value, we need to check how right it is for which we need the Loss function.**

Another thing to note is that the **random variable could be discrete or continuous.**

In the Grade case, the random variable would be discrete as we have 3 values for Grade(A, B, or C), for Height and Weight it would be continuous as the height could be anything like 120 cms. or 145 cms. and it could be all continuous values say from 50 cm to 250 cm. Similarly, weight is also a continuous quantity.

So, anything which takes on a real number is going to be continuous typically and anything which takes on only specific values(for example 0 or 1) would be a discrete random variable.

Grades



- A random variable can either take continuous values (for example, $weight$, $height$)
- Or discrete values (for example, $grade$, $nationality$)
- For this discussion we will mainly focus on discrete random variables

Marginal Distribution(Probability Distribution)

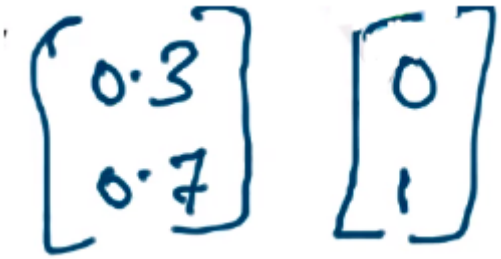
A distribution is nothing but a table like the below:

G	$P(G = g)$
A	0.1

C	0.7

So, the above is a distribution for a Random Variable Grade and it could take one of these values(A, B, C) and this distribution table tells: for every value that the random variable can take, the probability of the random variable taking on that value. And of course, all these values are going to sum to 1.

So, when we take the output of a Sigmoid Neuron, let's say Sigmoid Neuron tells the output is 0.7 if we think of this as a distribution, the probability of output being '1'(or belonging to class 1) is 0.7 and the probability of output being '0'(belonging to class 0) is 0.3. So, this is the distribution that Sigmoid Neuron gives. And we also have the true output as a distribution, suppose that image contains text then the true output is 1 and true distribution would give the probability of image containing text as 1 and probability of image not containing text as 0.



We can represent these distributions as:

True Distribution: [0 1]

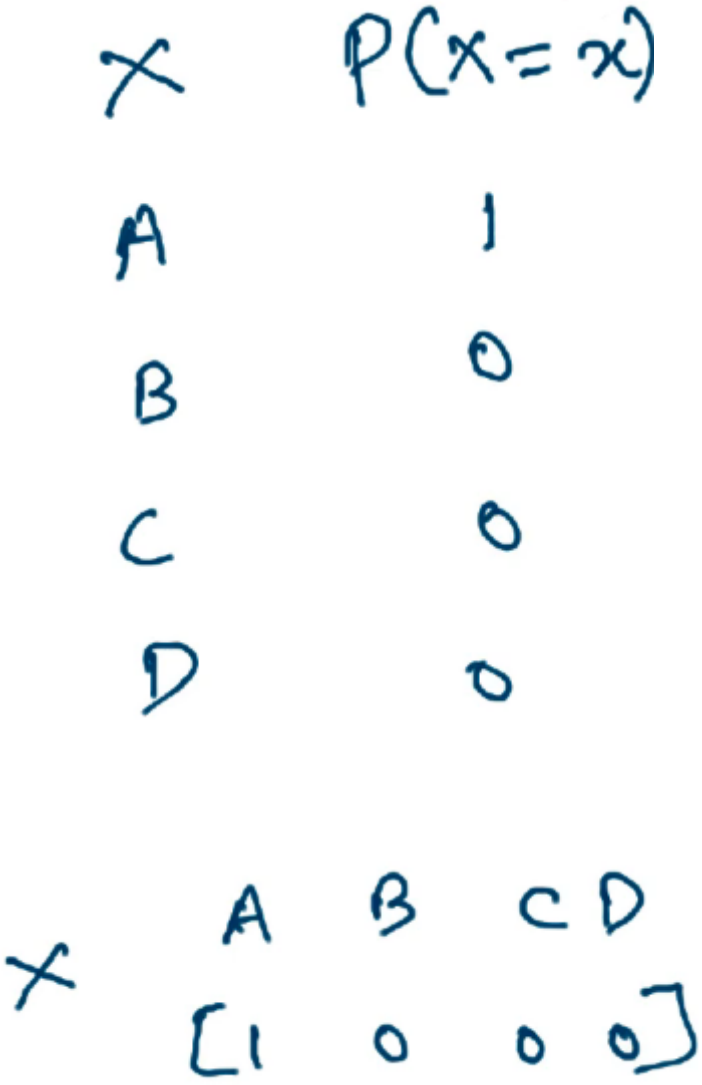
Predicted Distribution: [0.3 0.7]

As these two distributions are just two vectors, squared error loss can be used in this situation but by doing so, we are actually ignoring a few things like the fact this is actually a probability distribution, it has certain properties like all these values are greater than equal to 0 and less than 1 and all the values for a distribution would sum up to 1. So, we will look at a loss function that can deal with the quantities that are probabilities.

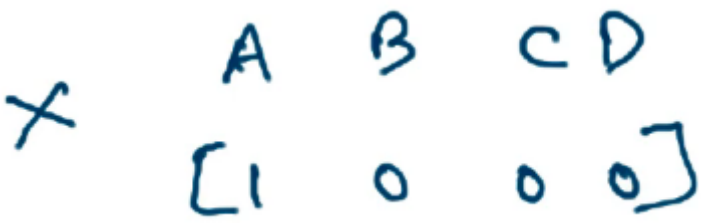
Certain Events:

So, at this point, there is no question of the probability of team D winning or C winning or B winning because we know that A has actually won the game. So, for **certain events**(means sure events or events that happen with 100% probability) as well, we can still write it as a distribution.

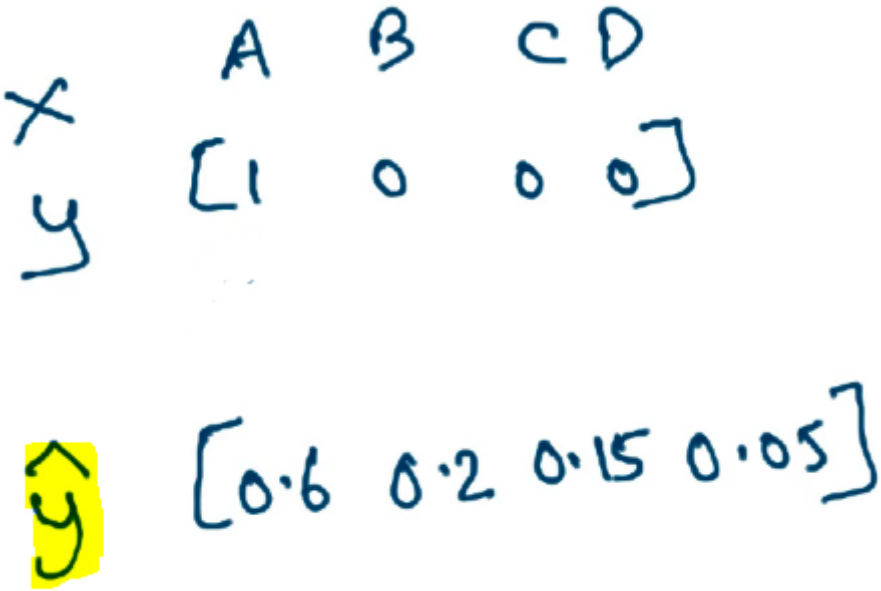
Let' say we have this Random Variable X which indicates which team has actually won the tournament and it can take on 4 values A, B, C and D. And now we can ask the probability of a particular team winning, and now since the event has already happened, we can write the following as the true distribution where all the probability mass is focussed on the certain or sure event:



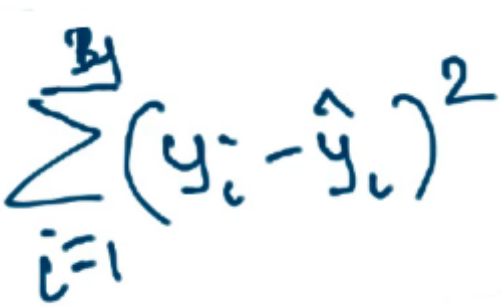
So, for any kind of distribution, we need to give values for all possible outcomes and it is possible that some of these outcomes have 0 probability mass.



Now if you someone(who watched a few games of this tournament) about what he thinks about the outcome of the tournament and he tells you this the distribution as in the below image(yellow highlighted):



And wants to know how close his prediction was to the true output, so know this we can use the Squared Error Loss:



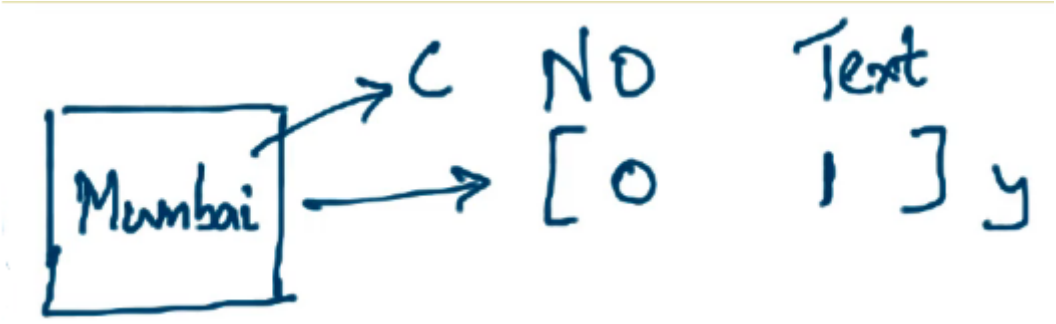
loss function which takes into account the thing that these distribution values are probabilities.

Why do we care about distributions?

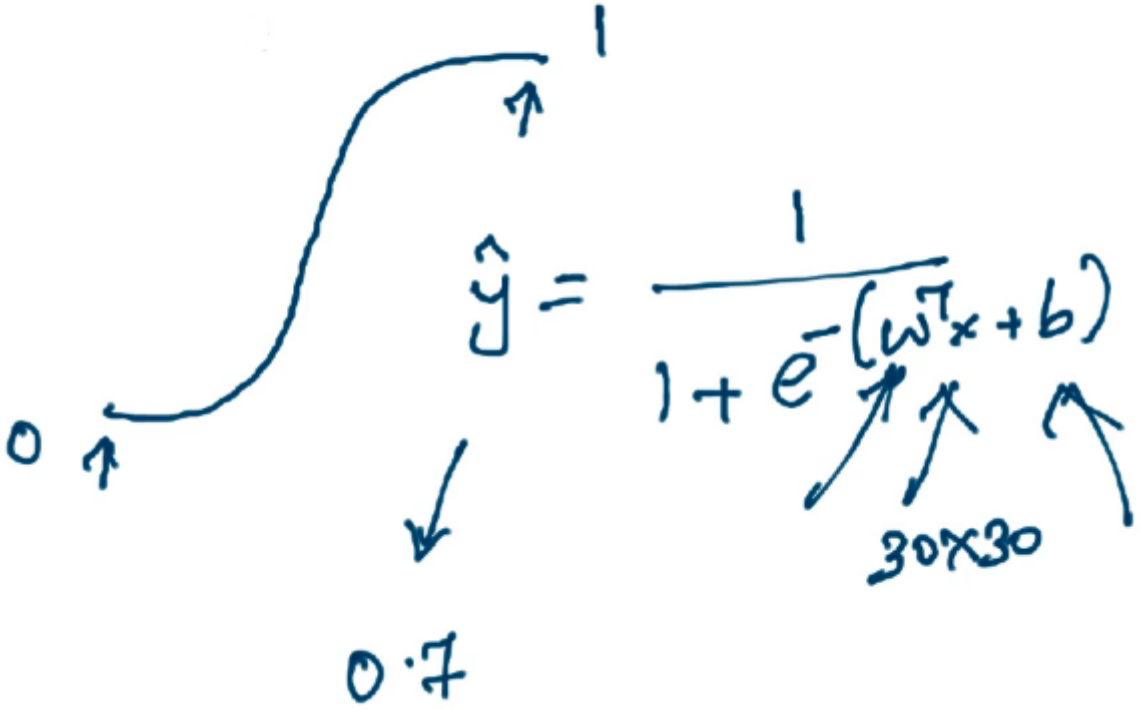
We will be given an image and our first task is to find out if it contains text or not.

At training time, we could think like there is some random variable associated here that tells **Class** and the class could be 0(means it does not contains text) or 1(means it contains text).

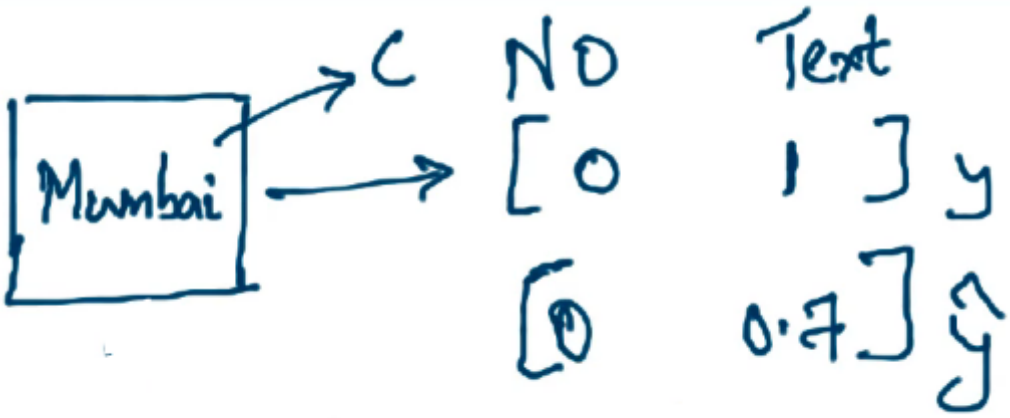
So, for the above case(when the text in the input is Mumbai), we can think that all the probability mass is on the event that the image contains 'Text' and the probability mass on the event 'No Text' is 0 because this is a certain event, we have already seen the image, it contains text so there is no probability here for the 'No Text' event, its certain that the image contains text, we could write it as below distribution(here as **y equals [0 1]**, this is known as one-hot encoded form as only one of the entries is 1 and rest all are 0):



At the training time, we are using a Sigmoid function(gives value between 0 and 1) and we want the output of the Sigmoid to be 1 when the image contains text and the output of the Sigmoid should be 0 if the image does not contain text. And the model



Here x in the above equation represents the input image which could be of size say 30 X 30, so we have 900 values in x, and corresponding we have 900 weights and then bias terms; using all of this we compute the predicted output. Suppose it gives a value of **0.7**. Now again, we can think of this as a probability distribution, it tells the probability of the image containing text as 0.7 and the remaining i.e 1–0.7 = 0.3 as the probability that the image does not contains text. Ideally, if the model was perfect, it should have given an output of 1 and the probability of the image not containing text as 0 but the model is not perfect, its parameters are still being trained and the output we get at this point is 0.7.



And now we are interested in knowing how bad is the model in the current parameter setting so that we can update the weights accordingly to make it even better and slowly reach the distribution where the loss is 0.

We can use the squared error loss for this. We will discuss another loss function for this.

In the above scenario, we have discussed the Binary classification but the same concept holds for Multi-class classification as well.

Machine Learning Probability Theory Artificial Intelligence Deep Learning Padhai